

# Configuration Manual

MSc Research Project  
MSc Cyber Security

**Sidhant Prakash Patil**  
Student ID: x23128577

School of Computing  
National College of Ireland

Supervisor: Prof Liam McCabe

National College of Ireland  
MSc Project Submission Sheet  
School of Computing



**Student Name:** Sidhant Prakash Patil  
**Student ID:** X23128577  
**Programme:** MSc cyber Security **Year:** 2024  
**Module:** MSc Research Project  
**Lecturer:** Prof Liam Mccabe  
**Submission Due Date:** 12/12/2024  
**Project Title:** Configuration Manual  
**Word Count:** 917..... **Page Count: 19**.....

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

**Signature:** Sidhant Patil.....

**Date:** 12/12/2024.....

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
<b>Attach a Moodle submission receipt of the online project submission</b> , to each project (including multiple copies).	<input type="checkbox"/>
<b>You must ensure that you retain a HARD COPY of the project</b> , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

<b>Office Use Only</b>	
Signature:	
Date:	
Penalty Applied (if applicable):	

# Configuration Manual

Sidhant Patil  
Student ID: x23128577

## 1 Introduction

The configuration manual provides detailed instructions of configuration, setup, and operation of the Blockchain based EHR systems. Ethereum blockchain is used with smart contracts written in solidity, IPFS for decentralized storage and React.js for creating a simple scalable, interactive frontend.

## 2 System configuration

Hardware of the system.

- Processor: Ryzen 9 6900H
- Operating system: Windows 11 Home
- Storage: 1 TB Gen 4 SSD
- RAM: 16 GB 6400Mhz

Versions of Software used.

- **Truffle Suite:** Truffle v5.11.5 (core: 5.11.5)
- **Web3:** Web3.js v1.10.0
- **Ganache:** UI Ganache v2.7.1
- **MetaMask:** 12.6.2 (browser extension)
- **IPFS Desktop:** CLI V 0.31.0
- **Visual Studio Code :** Version: 1.95.3 (user setup)
- **React.js:** version 18.2.0
- **Solidity:** Compiler version 0.8.19
- **Node.js:** Version 20.17.0
- **NPM (Node Package Manager):** version 10.8.3

## 3 Setup and configure development environment

This section provides instructions to set up the environment and configuration of the system components.

- 3.1 Install Node.js and NPM from <https://nodejs.org/en/download/package-manager> and verify the installation by running node -v or npm -v in the command prompt.



```
Windows PowerShell
PS C:\Users\siddh> npm -v
10.8.3
PS C:\Users\siddh> node -v
v20.17.0
PS C:\Users\siddh> |
```

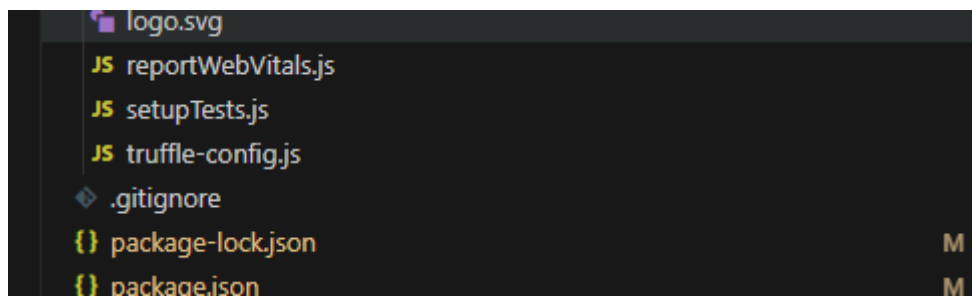
- 3.2 Setup git to track and upload the files to save changes made to the project. Install and verify the git using git --version command.

```
PS C:\Users\siddh> git --version
git version 2.47.0.windows.1
PS C:\Users\siddh> |
```

- 3.3 Install Truffle globally to compile and deploy the smart contracts by using the npm install -g truffle and verify the installation using truffle version command

```
PS C:\Users\siddh> truffle version
Truffle v5.11.5 (core: 5.11.5)
Ganache v7.9.1
Solidity v0.5.16 (solc-js)
Node v20.17.0
Web3.js v1.10.0
PS C:\Users\siddh> |
```

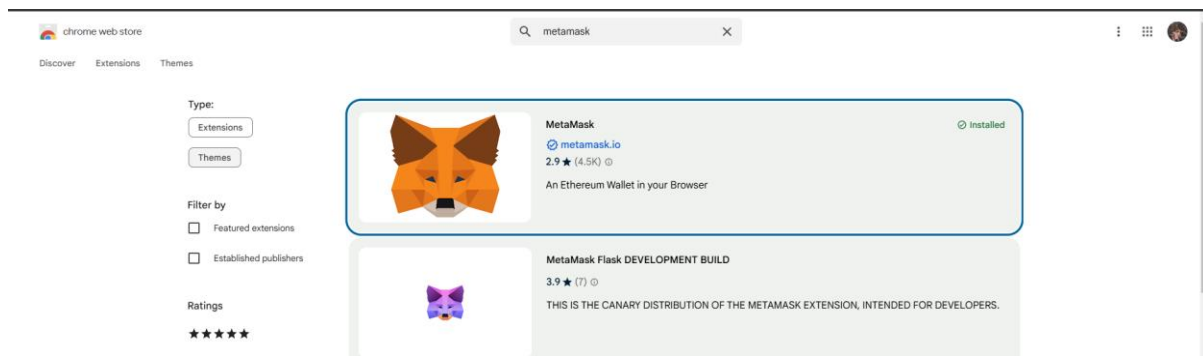
- 3.4 Download and install Ganache from the official website. launch ganache and add truffle project by selecting the truffle.config file created after installing truffle.









- 3.5 Configure the RPC server URL to HTTP://127.0.0.1:7545 and network ID to 1337 from 5777 as 1337 is a default port used by ganache even if it shows 5777.

ADDRESS	BALANCE	TX COUNT	INDEX
0xF63Df8616B048E85c7F2263E769AF3Bf5AD0E66	100.61 ETH	139	0

### 3.6 Add and configure MetaMask extension on preferred browser.



### 3.7 Import 3 prefunded accounts form ganache for testing by copying the private key provided in ganache.

ADDRESS 0×F63DfF8616B048E85c7F2263E769AF3Bf5AD0E66	BALANCE 100.23 ETH	TX COUNT 259	INDEX 0	
ADDRESS 0×F75Da7924f3Ed473F949c4A63Ee7238e9632A60B	BALANCE 98.95 ETH	TX COUNT 116	INDEX 1	
ADDRESS 0×ff21003959e23E6F4eeB2b8f3B4a69690615Eb04	BALANCE 99.99 ETH	TX COUNT 21	INDEX 2	
ADDRESS 0×645277355E787f13163BeF912099379689FFd217	BALANCE 100.00 ETH	TX COUNT 4	INDEX 3	
ADDRESS 0×6175B7fcB5c09Bfe12CF8Cca2B31926d9B64D0Bb	BALANCE 100.00 ETH	TX COUNT 6	INDEX 4	
ADDRESS 0×8f0cdC93a7635E07ed7AD1D5050C8843B0396B2A	BALANCE 100.00 ETH	TX COUNT 5	INDEX 5	
ADDRESS	BALANCE	TX COUNT	INDEX	

### 3.8 Copy Private key

**ACCOUNT INFORMATION**

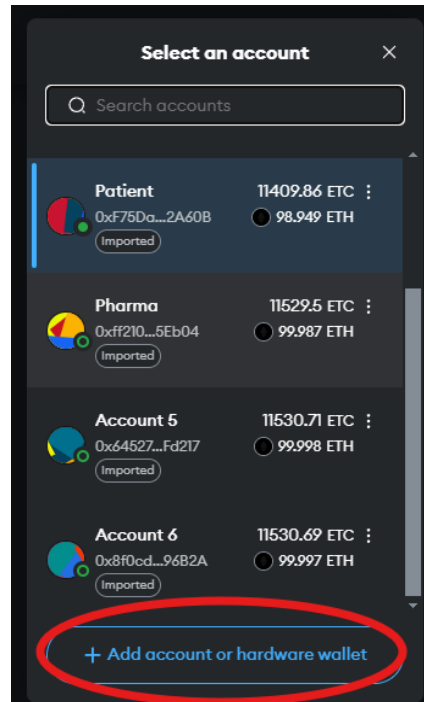
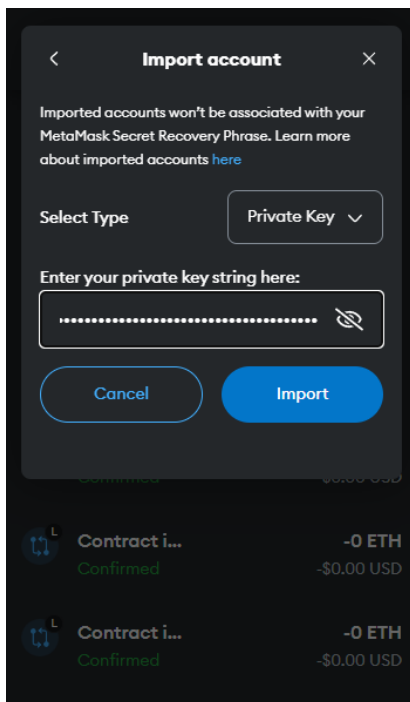
**ACCOUNT ADDRESS**  
0×F63DfF8616B048E85c7F2263E769AF3Bf5AD0E66

**PRIVATE KEY**  
0×d558d726b88c476366448852f62342a4a98cec97d82f530c409202d4f4833652

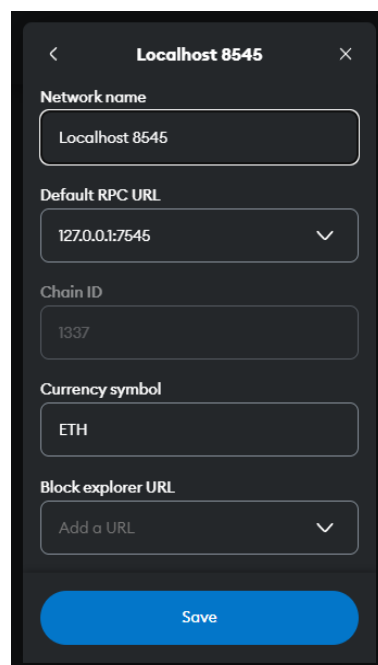
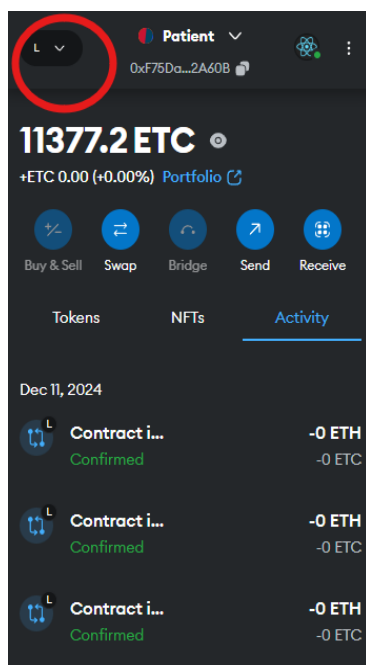
**Do not use this private key on a public blockchain; use it for development purposes only!**

**DONE**

### 3.9 Import accounts to MetaMask

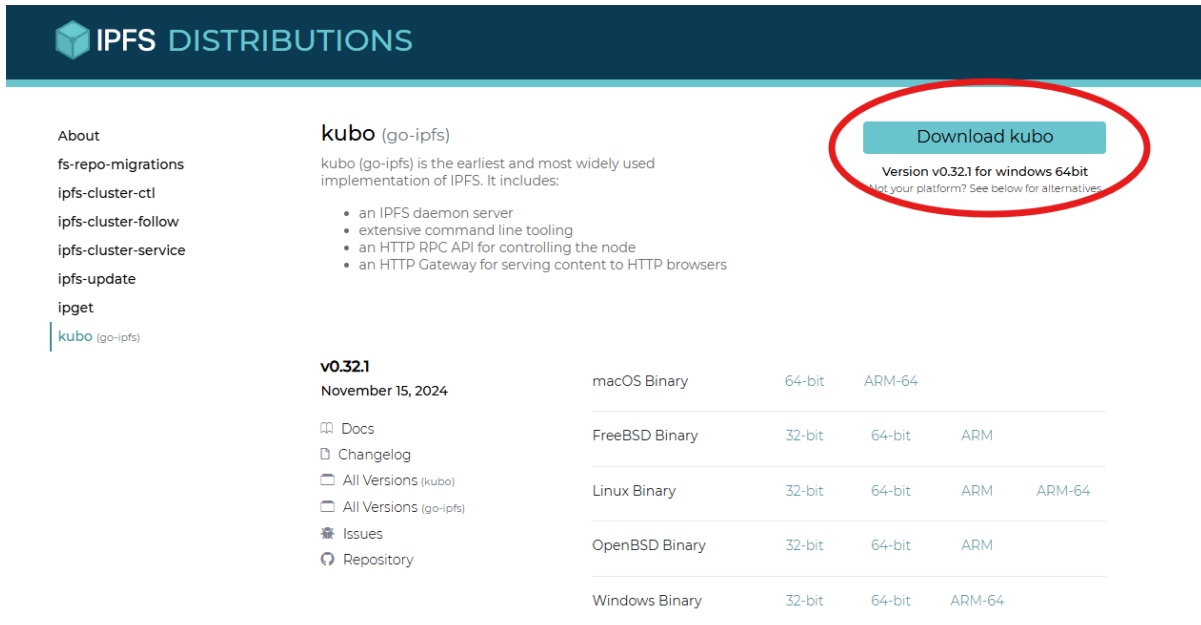


- 3.10 Add a custom network with network name: Ganache Network or the name suggested by the MetaMask to avoid any conflicts within the networks, RPC URL: HTTP://127.0.0.1:7545 and Chain ID: 1337 (note: as you can see the chain id is 1337 by default for local ganache network even if it shows 5777 in the ganache application)



- 3.11 Setup and install IPFS Kubo as guided in the provided link <https://docs.ipfs.tech/install/command-line/> for storing the records. follow below steps for windows.

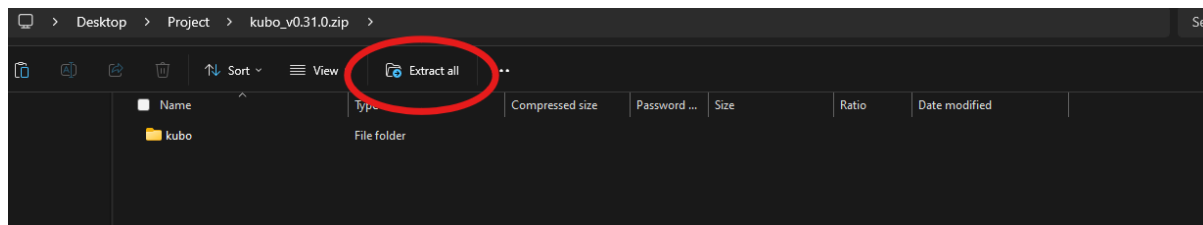
3.12 Download the binary for windows from <https://dist.ipfs.tech/#kubo> or use `wget https://dist.ipfs.tech/kubo/v0.32.1/kubo_v0.32.1_windows-amd64.zip -Ofile kubo_v0.32.1.zip` command



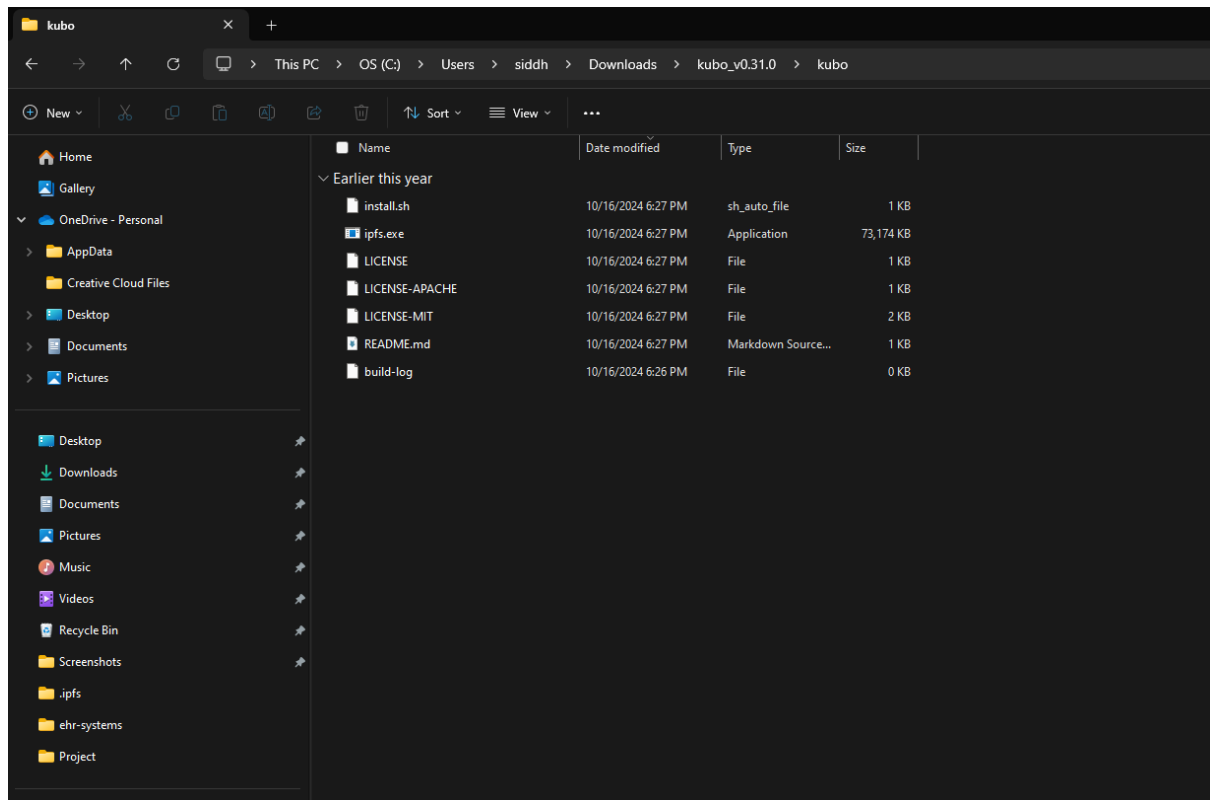
The screenshot shows the IPFS Distributions website for kubo. On the left is a sidebar with links: About, fs-repo-migrations, ipfs-cluster-ctl, ipfs-cluster-follow, ipfs-cluster-service, ipfs-update, ipget, and kubo (selected). The main content area for kubo (go-ipfs) includes a description, a list of features (IPFS daemon server, command line tooling, HTTP RPC API, HTTP Gateway), and a table of binaries. A red circle highlights the 'Download kubo' button, which specifies 'Version v0.32.1 for windows 64bit'.

Platform	32-bit	64-bit	ARM	ARM-64
macOS Binary		64-bit		ARM-64
FreeBSD Binary	32-bit	64-bit	ARM	
Linux Binary	32-bit	64-bit	ARM	ARM-64
OpenBSD Binary	32-bit	64-bit	ARM	
Windows Binary	32-bit	64-bit		ARM-64

3.13 Extract the zip file at preferred location by extractor like 7zip or by using command `Expand-Archive -Path kubo_v0.32.1.zip -DestinationPath ~\Apps\kubo_v0.32.1`



3.14 Redirect to the folder where the zip is extracted: `cd /destination` to the folder



### 3.15 Check the IPFS.exe and verify version

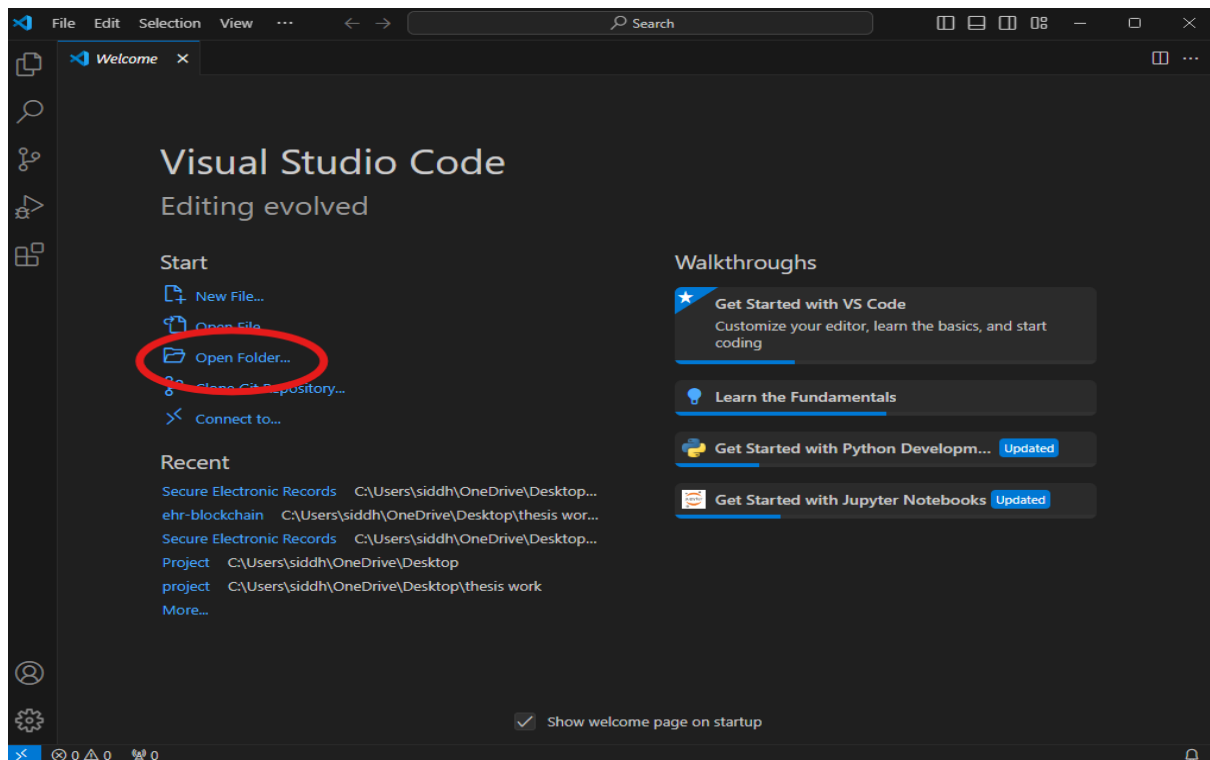
```
Windows PowerShell
PS C:\Users\siddh\downloads\kubo_v0.31.0> cd kubo
PS C:\Users\siddh\downloads\kubo_v0.31.0\kubo> .\ipfs.exe --version
ipfs version 0.31.0
PS C:\Users\siddh\downloads\kubo_v0.31.0\kubo> |
```

### 3.16 Install or Update Visual Studio Code from <https://code.visualstudio.com/download>

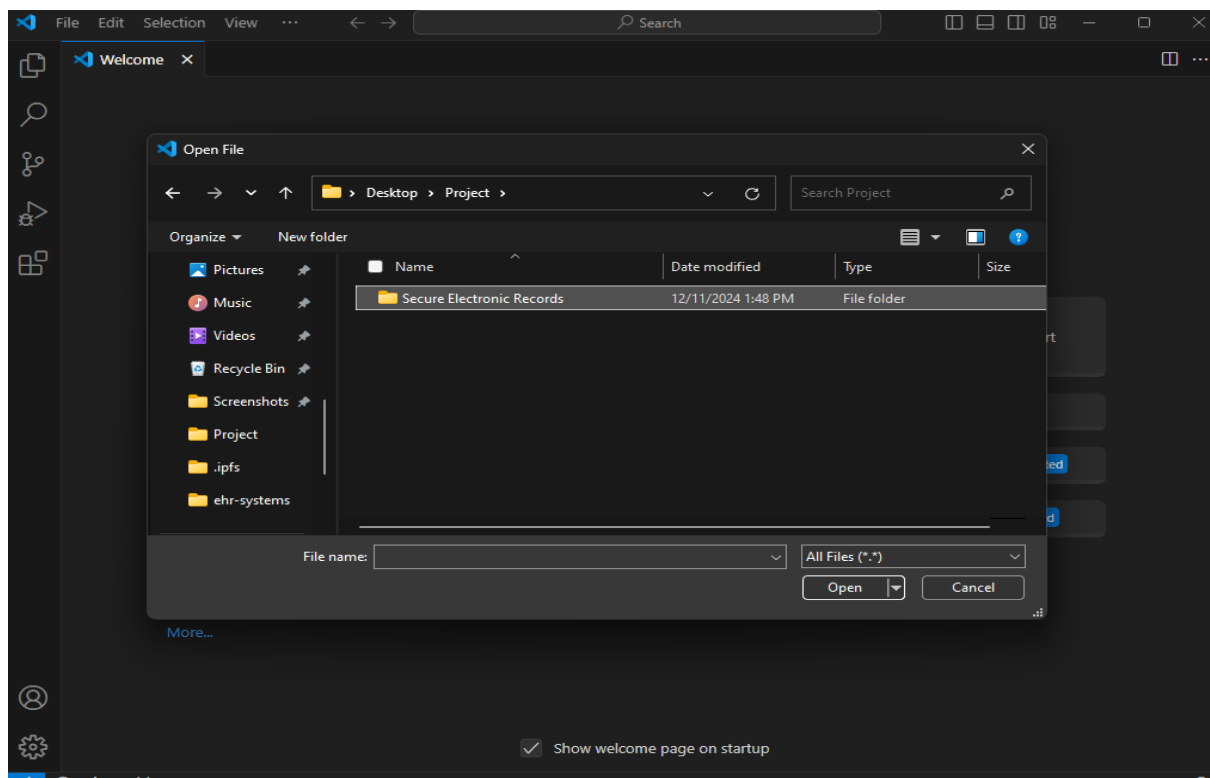




3.17 Open the folder with project files or by cloning the repository using 'git clone repo url' and 'cd Project folder'

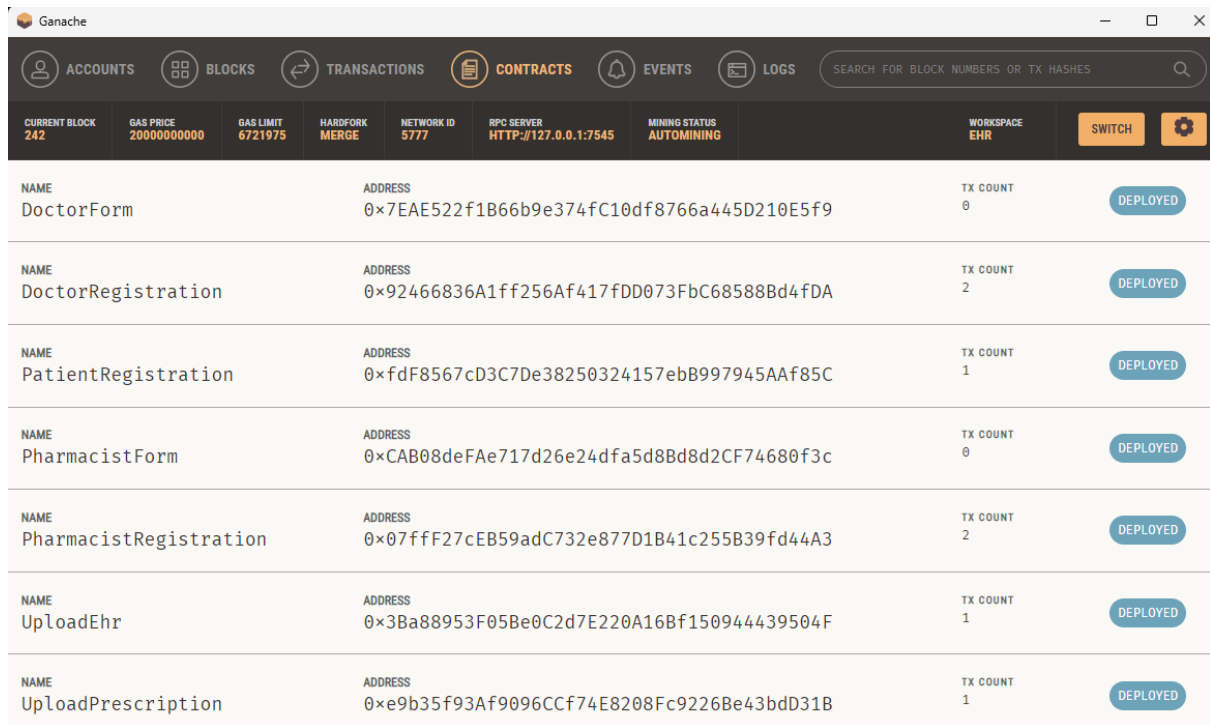


3.18 Select the project directory and click open





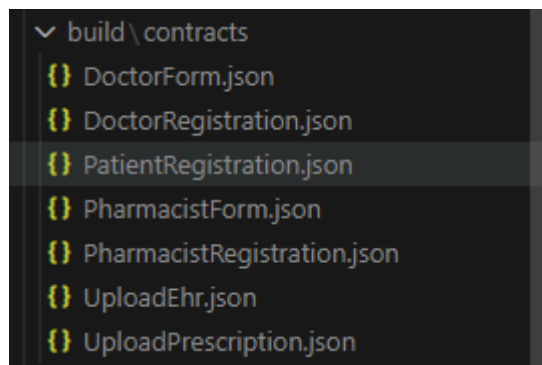
### 3.22 Verify the deployment of the contracts in ganache.



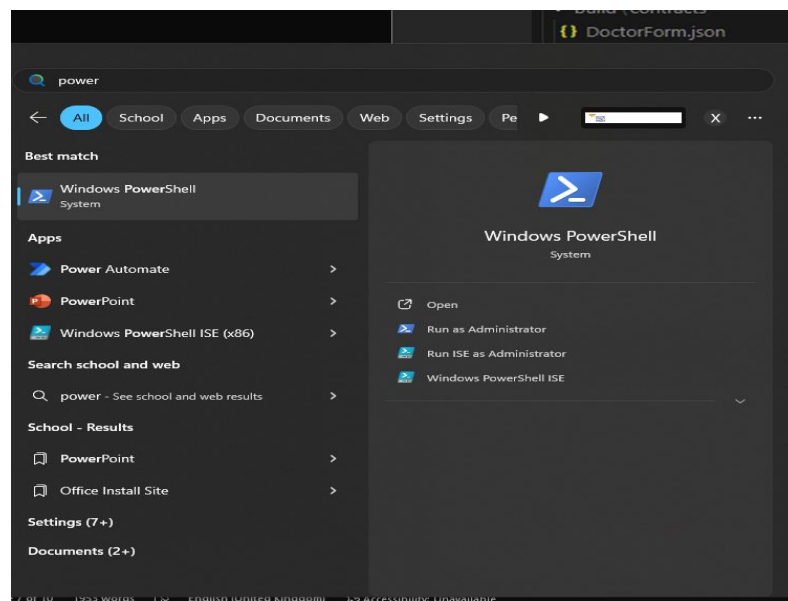
The screenshot shows the Ganache application window. The top navigation bar includes tabs for ACCOUNTS, BLOCKS, TRANSACTIONS, CONTRACTS (which is active), EVENTS, and LOGS. Below the navigation bar, a status bar displays various network metrics: CURRENT BLOCK 242, GAS PRICE 20000000000, GAS LIMIT 6721975, HARDFORK MERGE, NETWORK ID 5777, RPC SERVER HTTP://127.0.0.1:7545, MINING STATUS AUTOMINING, and WORKSPACE EHR. A search bar is also present. The main area displays a table of deployed contracts.

NAME	ADDRESS	TX COUNT	
DoctorForm	0x7EAE522f1B66b9e374fC10df8766a445D210E5f9	0	DEPLOYED
DoctorRegistration	0x92466836A1ff256Af417fDD073FbC68588Bd4fDA	2	DEPLOYED
PatientRegistration	0xfdF8567cD3C7De38250324157ebB997945AAf85C	1	DEPLOYED
PharmacistForm	0xCAB08deFAe717d26e24dfa5d8Bd8d2CF74680f3c	0	DEPLOYED
PharmacistRegistration	0x07fff27cEB59adC732e877D1B41c255B39fd44A3	2	DEPLOYED
UploadEhr	0x3Ba88953F05Be0C2d7E220A16Bf150944439504F	1	DEPLOYED
UploadPrescription	0xe9b35f93Af9096CCf74E8208Fc9226Be43bdD31B	1	DEPLOYED

### 3.23 Verify the abi files in build/contracts folder



- 3.24 Start the IPFS daemon open windows powershell from windows Search window and search powershell or press Win+R and type powershell and enter

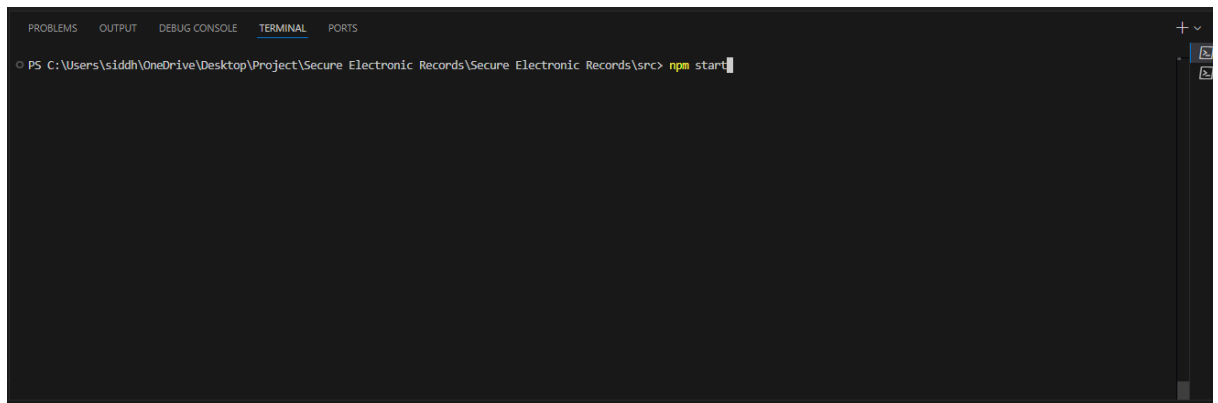


- 3.25 Enter command 'ipfs daemon'

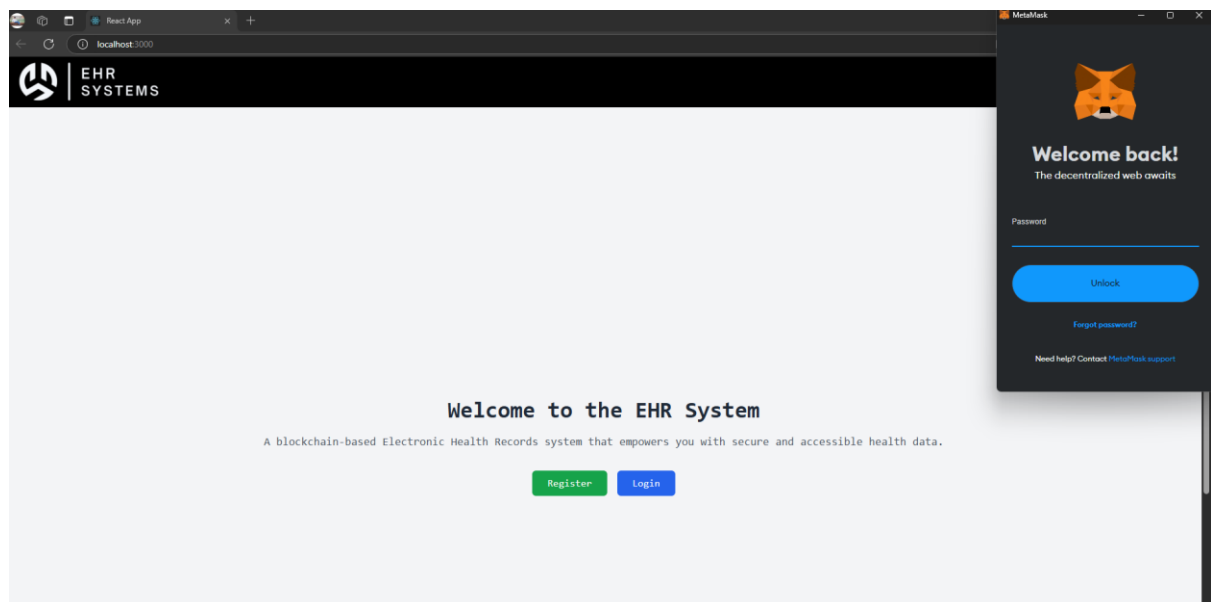
```
Windows PowerShell
PS C:\Users\siddh> ipfs daemon
Initializing daemon...
Kubo version: 0.31.0
Repo version: 16
System version: amd64/windows
Golang version: go1.23.2
PeerID: 12D3KoolWR4ECJyCNDcPv6USHMrk33QfJSK5A1r4xAsDrwB5ZKE1p
Swarm listening on 127.0.0.1:4001 (TCP+UDP)
Swarm listening on 169.254.164.110:4001 (TCP+UDP)
Swarm listening on 169.254.242.200:4001 (TCP+UDP)
Swarm listening on 169.254.35.190:4001 (TCP+UDP)
Swarm listening on 172.22.176.1:4001 (TCP+UDP)
Swarm listening on 192.168.0.253:4001 (TCP+UDP)
Swarm listening on 192.168.56.1:4001 (TCP+UDP)
Swarm listening on 192.168.56.2:4001 (TCP+UDP)
Swarm listening on [2a02:8084:31f:1300:6a3e:5b8:ab50:57a3]:4001 (TCP+UDP)
Swarm listening on [2a02:8084:31f:1300::4f]:4001 (TCP+UDP)
Swarm listening on [2a02:8084:31f:1300:a8cd:3f74:8ee7:d121]:4001 (TCP+UDP)
Swarm listening on [::1]:4001 (TCP+UDP)
Run 'ipfs id' to inspect announced and discovered multiaddrs of this node.
RPC API server listening on /ip4/127.0.0.1/tcp/5001
WebUI: http://127.0.0.1:5001/webui
Gateway server listening on /ip4/127.0.0.1/tcp/8080
Daemon is ready
2024-12-02T23:20:01.639Z ERROR cmd/ipfs kubo/daemon.go:1153
⚠ A NEW VERSION OF KUBO DETECTED

This Kubo node is running an outdated version (0.31.0).
13% of the sampled Kubo peers are running a higher version.
```

- 3.26 Start the project using 'npm start' command in visual studio code. This redirects to the dashboard on the default browser at <http://localhost:3000/>. MetaMask extension will pop up login and connect MetaMask to the website.



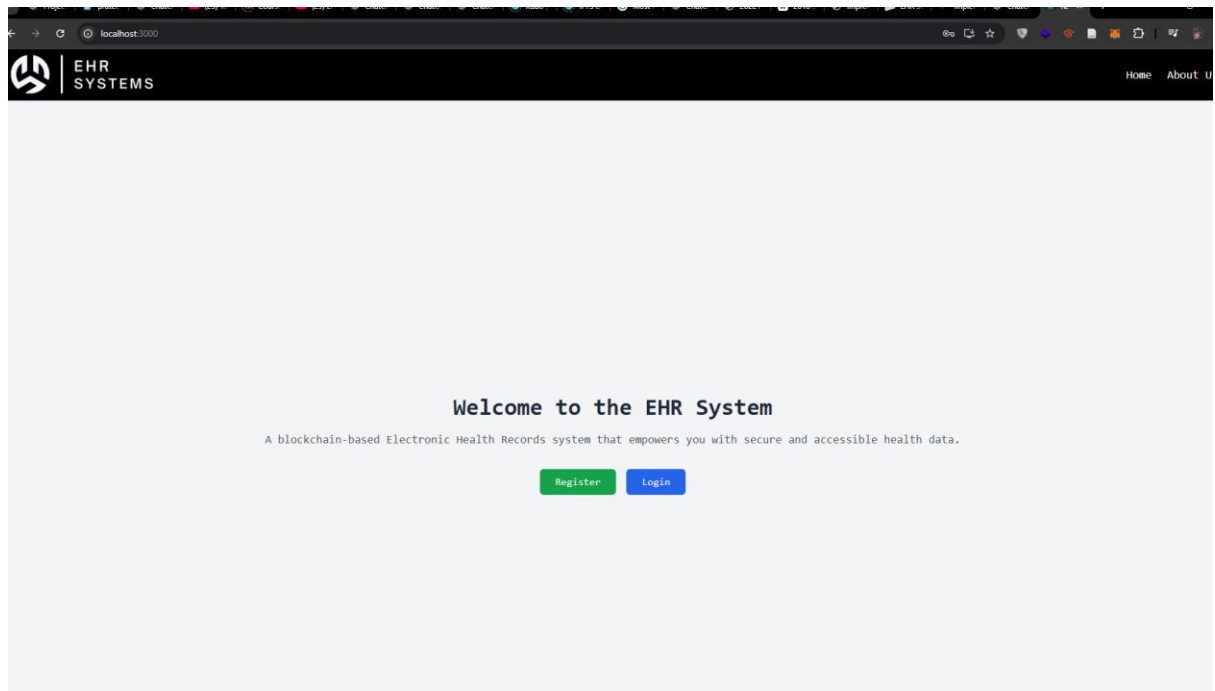
### 3.27 Application interface



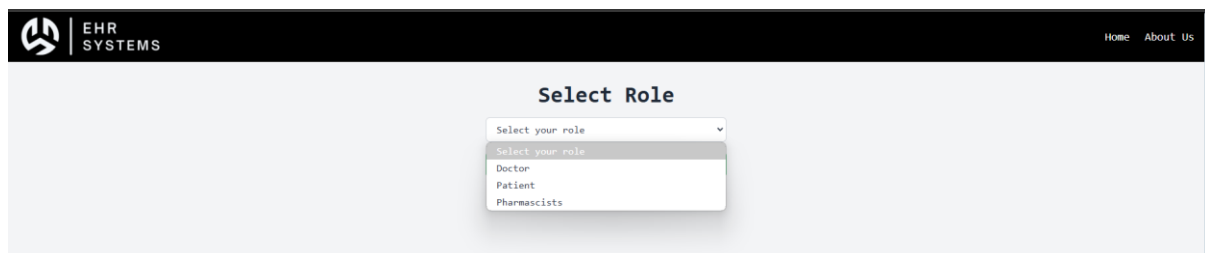
## 4 Testing the web Application

### 4.1 Register as a User:

#### 4.1.1 On the home page of the website click on register.



#### 4.1.2 Select a role (Patient, Doctor, Pharmacist).



#### 4.1.3 Register the user.

### Doctor Registration

<b>Wallet Public Address</b> <input type="text" value="Crypto Wallet's Public Address"/>	<b>Full Name</b> <input type="text" value="Enter Full Name"/>
<b>Hospital Name</b> <input type="text" value="Enter Hospital Name"/>	<b>Hospital Location</b> <input type="text" value="Enter Hospital Location"/>
<b>Date of Birth</b> <input type="text" value="mm/dd/yyyy"/>	<b>Gender</b> <input type="text" value="Select Gender"/>
<b>Email Address</b> <input type="text" value="123458"/>	<b>Create ID Number</b> <input type="text" value="Create ID Number"/>
<b>Specialization</b> <input type="text" value="Select Specialization"/>	<b>Department</b> <input type="text" value="Select Department"/>
<b>Designation</b> <input type="text" value="Select Designation"/>	<b>Work Experience (Years)</b> <input type="text" value="Enter Work Experience in Years"/>
<b>Password</b> <input type="text" value="*****"/>	<b>Confirm Password</b> <input type="text" value="Confirm Password"/>

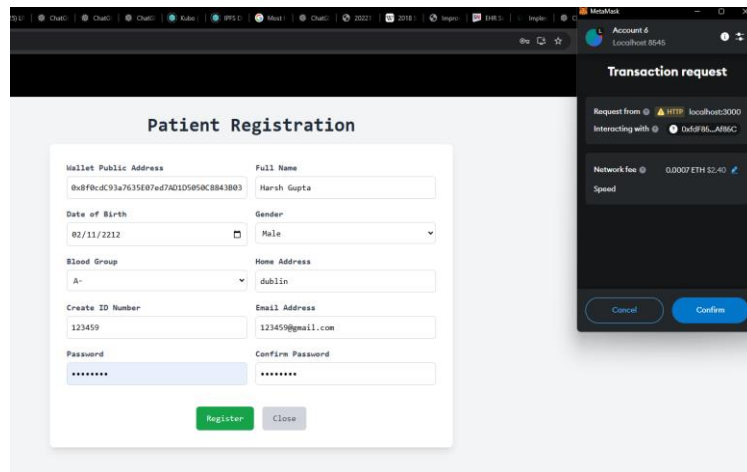
### Pharmacists Registration

<b>Wallet Public Address</b> <input type="text" value="Crypto Wallet Public Address"/>	<b>Pharmacy Name</b> <input type="text" value="Enter Pharmacist Center Name"/>
<b>Hospital Name</b> <input type="text" value="Enter Hospital Name"/>	<b>Location</b> <input type="text" value="Enter Pharmacist Center Location"/>
<b>Email Address</b> <input type="text" value="123458"/>	<b>Create ID Number</b> <input type="text" value="ID Number"/>
<b>Password</b> <input type="text" value="*****"/>	<b>Confirm Password</b> <input type="text" value="Confirm your Password"/>

### Patient Registration

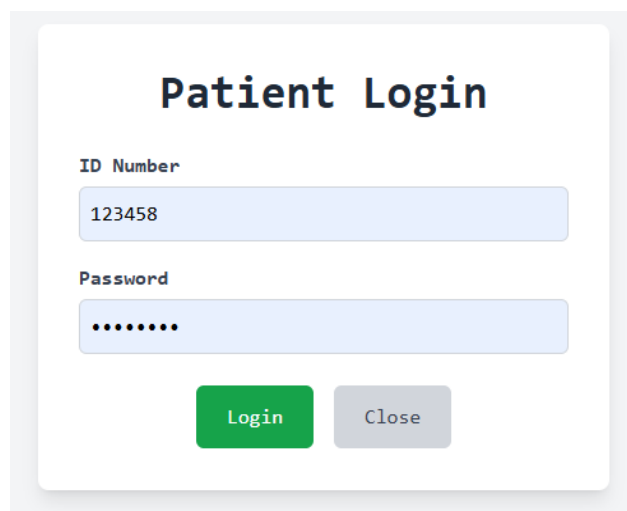
<b>Wallet Public Address</b> <input type="text" value="Crypto Wallet's Public Address"/>	<b>Full Name</b> <input type="text" value="Enter Full Name"/>
<b>Date of Birth</b> <input type="text" value="mm/dd/yyyy"/>	<b>Gender</b> <input type="text" value="Select Gender"/>
<b>Blood Group</b> <input type="text" value="Select Blood Group"/>	<b>Home Address</b> <input type="text" value="Enter your Permanent Address"/>
<b>Create ID Number</b> <input type="text" value="Create New ID Number"/>	<b>Email Address</b> <input type="text" value="123458"/>
<b>Password</b> <input type="text" value="*****"/>	<b>Confirm Password</b> <input type="text" value="Confirm your Password"/>

#### 4.1.4 Confirm the MetaMask transaction for registration.

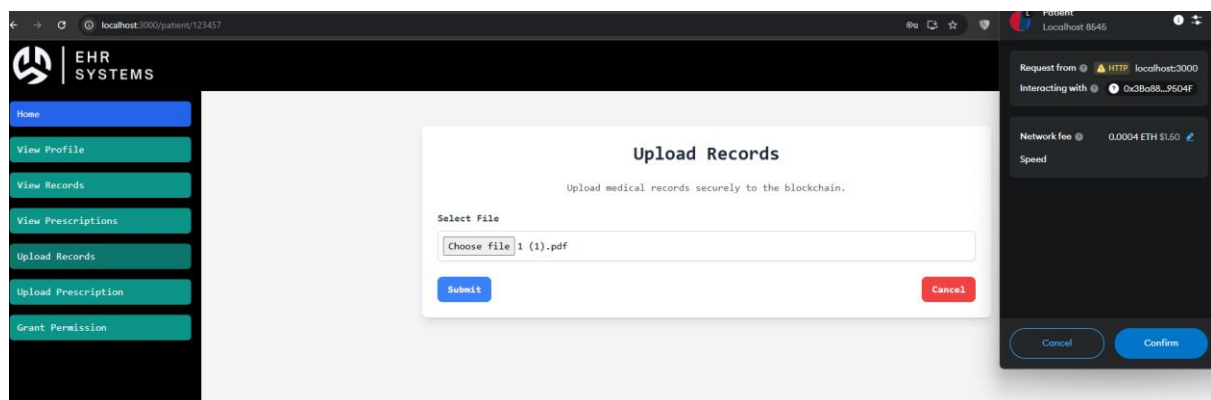


## 4.2 Upload Medical Records (Patients):

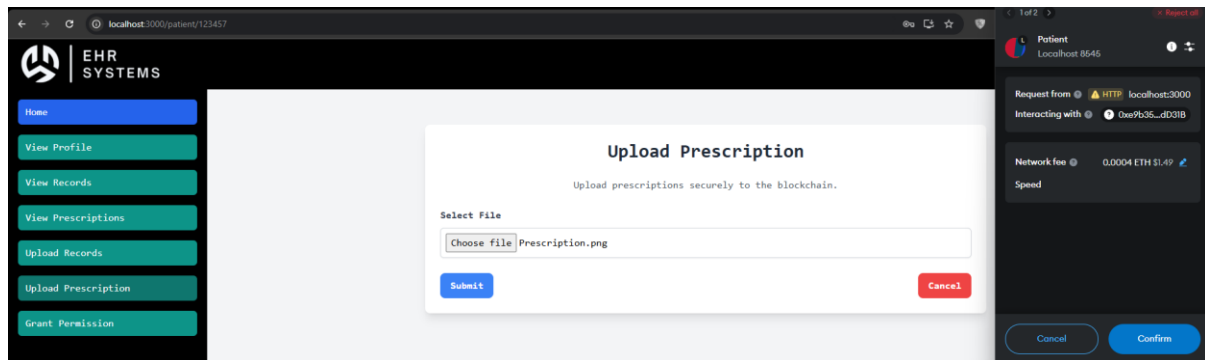
### 4.2.1 Login to the patient dashboard.



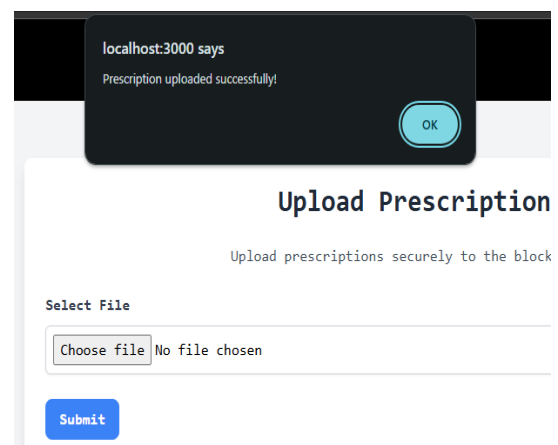
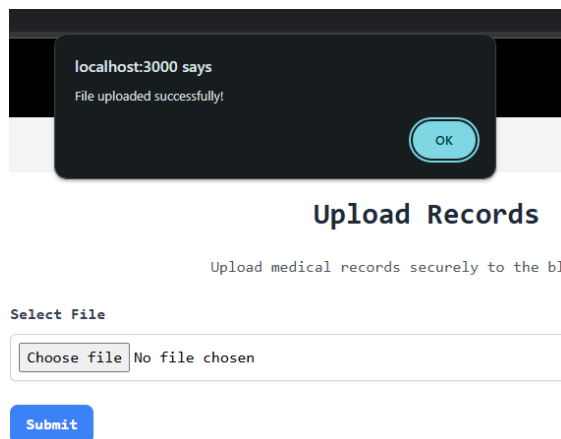
### 4.2.2 Upload the health records or prescription and confirm the transaction in MetaMask.





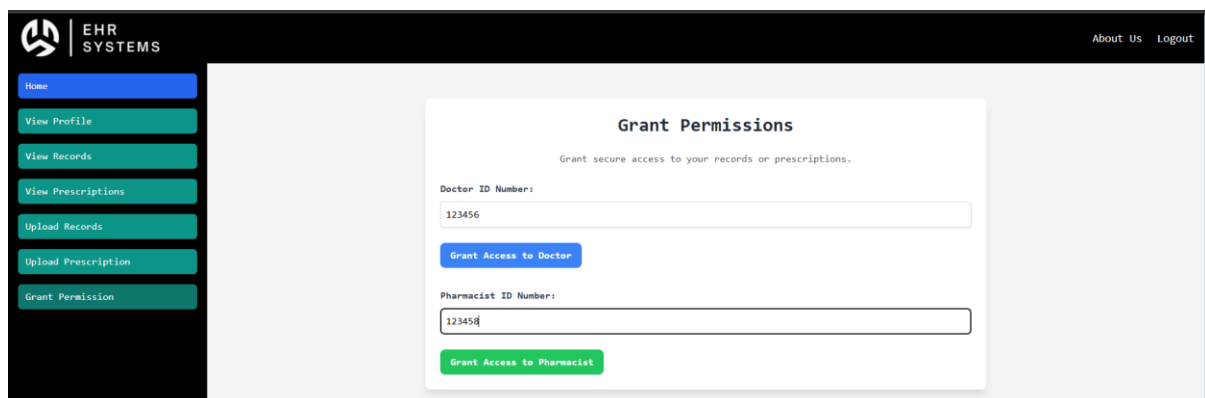


### 4.2.3 The file selected is encrypted and uploaded to IPFS.

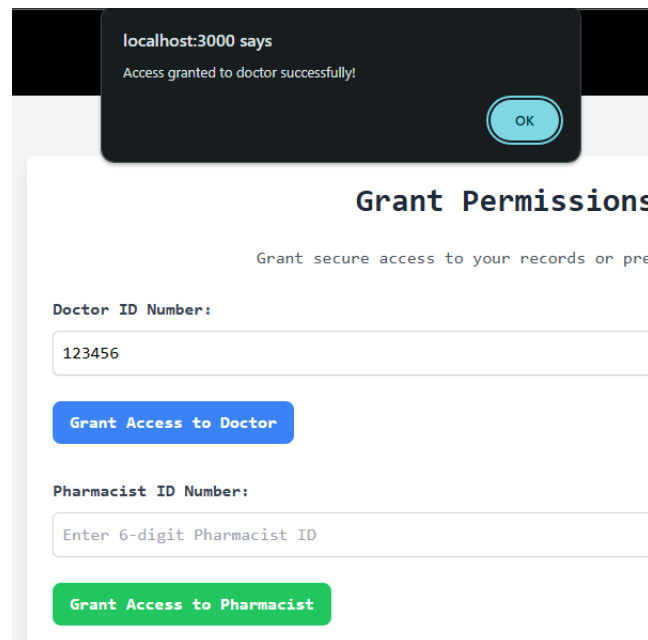
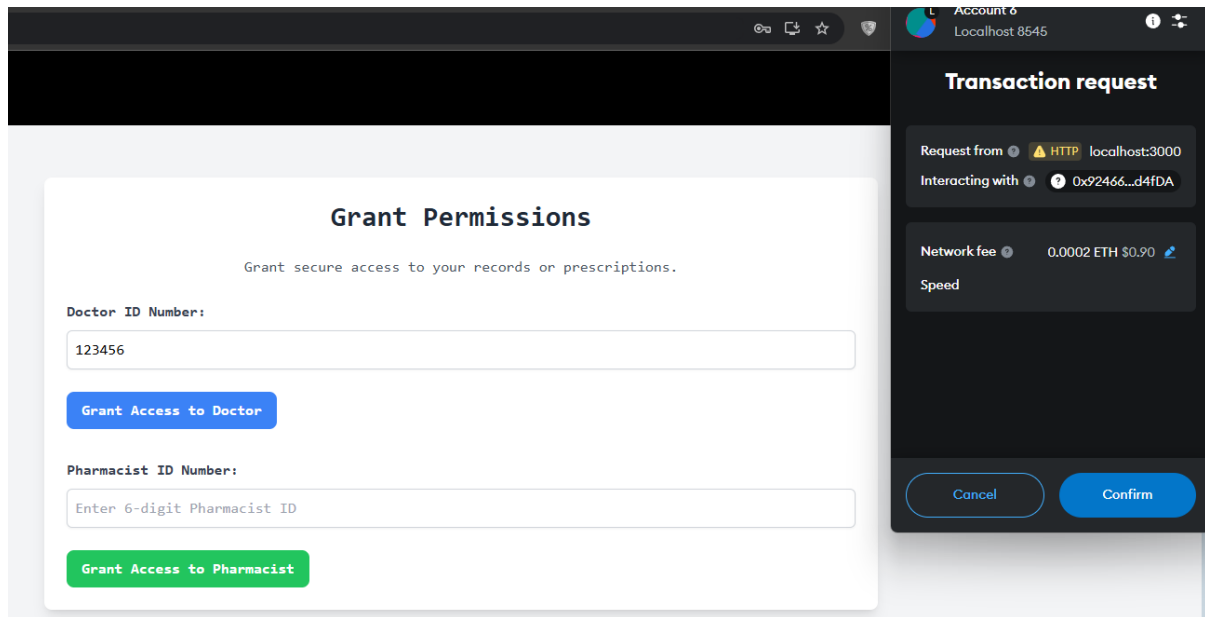


## 4.3 Grant Access to Doctors:

### 4.3.1 In the patient dashboard use grant permission tab to provide access to records and prescription uploaded.

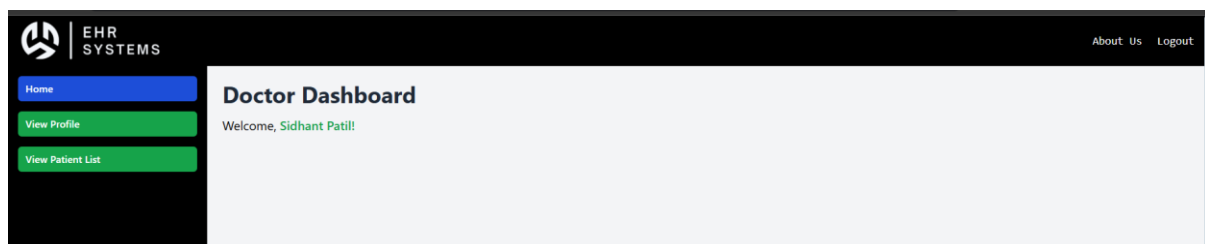


### 4.3.2 Confirm the transaction in MetaMask.

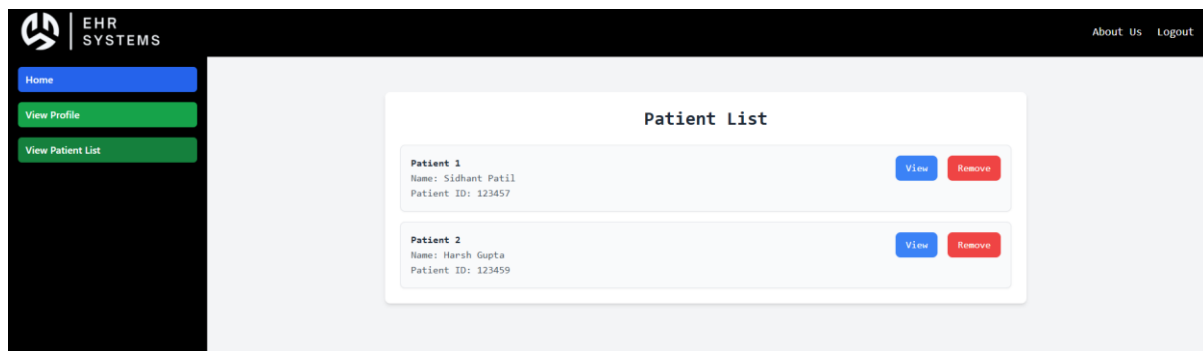


## 4.4 Access Records (Doctors):

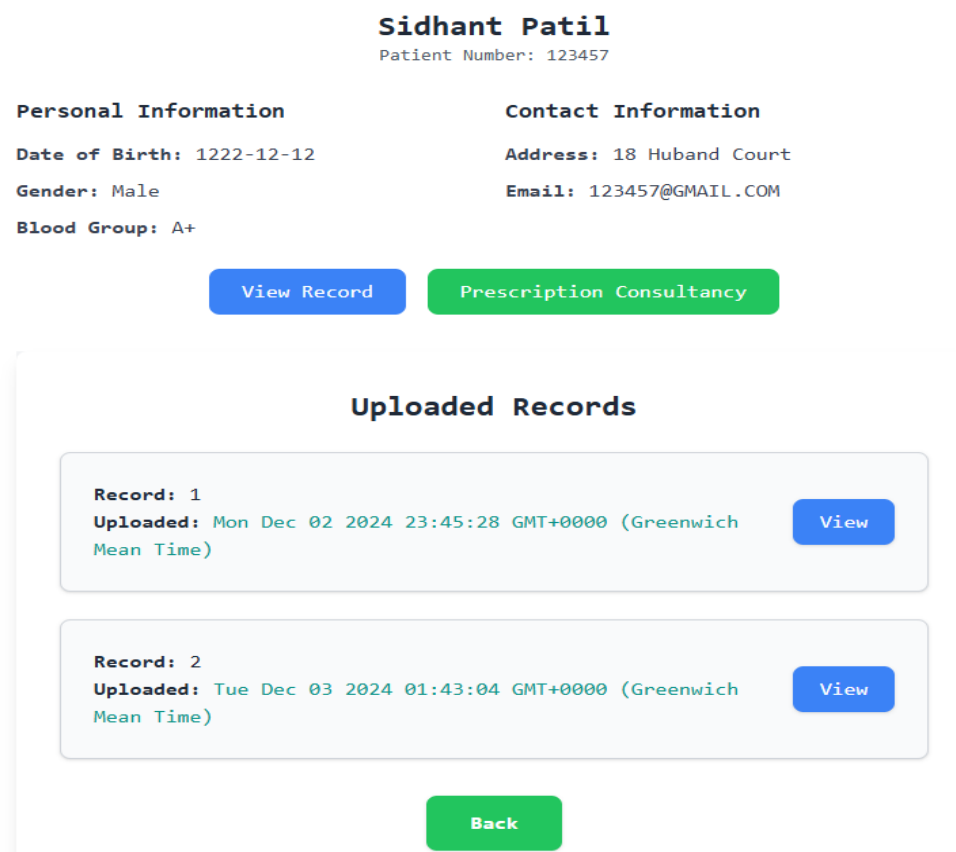
### 4.4.1 Log in as a doctor.



#### 4.4.2 View the list of patients who have granted access.



#### 4.4.3 View records in doctor dashboard and view prescription in pharmacist dashboard retrieves the medical records/prescription to which patient has given access.

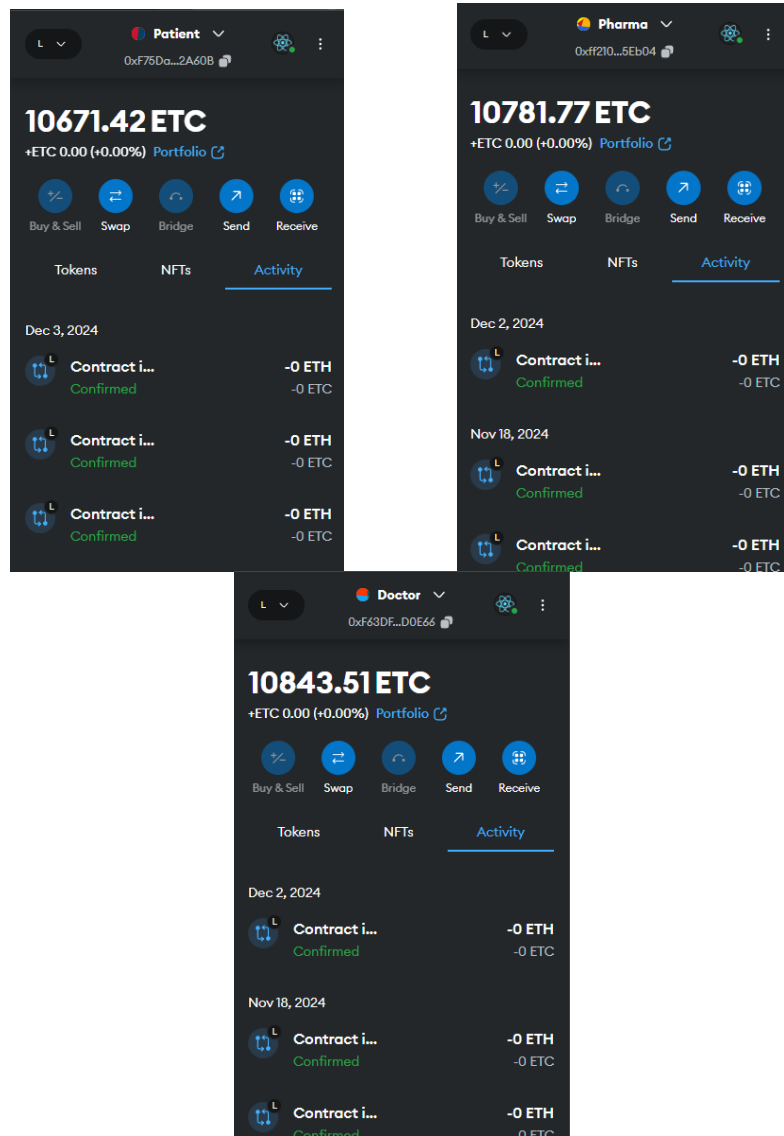


#### 4.5 Monitor Transactions in Ganache and MetaMask:

##### 4.5.1 Observe the transactions being recorded in Ganache.

Ganache									
ACCOUNTS	BLOCKS	TRANSACTIONS	CONTRACTS	EVENTS	LOGS	SEARCH FOR BLOCK NUMBERS OR TX HASHES			
CURRENT BLOCK 256	GAS PRICE 20000000000	GAS LIMIT 6721975	HARDFORK MERGE	NETWORK ID 5777	RPC SERVER HTTP://127.0.0.1:7545	MINING STATUS AUTOMINING	WORKSPACE EHR	SWITCH	
TX HASH <b>0x23c01b6d010b4620cca825622c61eb0997e6ea323af9552524e0077724dc3b58</b>									
FROM ADDRESS 0x8f0cdC93a7635E07ed7AD1D5050C8843B0396B2A									
TO CONTRACT ADDRESS 0x92466836A1ff256Af417fDD073FbC68588Bd4fDA									
GAS USED 99251									
VALUE 0									
TX HASH <b>0xf23b9b4b0f25211c5e604cff2d51f891906b4e4dea4350174642b93a983a9ea7</b>									
FROM ADDRESS 0x8f0cdC93a7635E07ed7AD1D5050C8843B0396B2A									
TO CONTRACT ADDRESS 0xe9b35f93Af9096CCf74E8208Fc9226Be43bdD31B									
GAS USED 181623									
VALUE 0									
TX HASH <b>0xd16cab7bf44d1454b3ed37fbd769b83ccfc6175d16558f7b300275f4a1080731</b>									
FROM ADDRESS 0x8f0cdC93a7635E07ed7AD1D5050C8843B0396B2A									
TO CONTRACT ADDRESS 0x38a88953F05Be0C2d7E220A16Bf150944439504F									
GAS USED 181645									
VALUE 0									
TX HASH <b>0x1ec3b452d15304e9cd894c1ada85e53218591dc8d85df3e2048d14e039006965</b>									
FROM ADDRESS 0xF75Da7924f3Ed473F949c4A63Ee7238e9632A60B									
TO CONTRACT ADDRESS 0xe9b35f93Af9096CCf74E8208Fc9226Be43bdD31B									
GAS USED 164523									
VALUE 0									
TX HASH <b>0x25f849628054e91917c1112e9e01e21d52b48c20786383633892e8c10ec8b2f4</b>									
FROM ADDRESS 0xF75Da7924f3Ed473F949c4A63Ee7238e9632A60B									
TO CONTRACT ADDRESS 0xe9b35f93Af9096CCf74E8208Fc9226Be43bdD31B									
GAS USED 164523									
VALUE 0									

## 4.5.2 Check account balances and transaction history in MetaMask.



## 5 Troubleshooting

### Common Issues faced and Solutions

1. **MetaMask and Ganache connectivity:** Verify the MetaMask local network configuration matches the Ganache network settings. Try launching new ganache project with different RPC URL port. To verify the connection make a transaction between the accounts.
2. **Smart Contract related issues:** Check for any syntax errors in code any overshadowing. Reconfigure the truffle-config.js file to match the port, network\_id and verify the solidity compiler version. Make sure the correct Abi is called where required. Verify ganache is up and running.
3. **IPFS Connection Issues:** Verify the IPFS gateway address and Port as it may be used by other application.

### 9. References

1. Truffle: <https://archive.trufflesuite.com/>
2. Web3.js: <https://web3js.readthedocs.io/en/v1.10.0/>
3. Ganache: <https://archive.trufflesuite.com/ganache/>
4. MetaMask: <https://metamask.io/>

5. IPFS: <https://github.com/ipfs/ipfs-desktop>
6. Visual Studio Code: <https://code.visualstudio.com/>
7. React.js: <https://react.dev/>
8. Solidity: <https://soliditylang.org/>
9. Node.js: <https://nodejs.org/en>
10. NPM: <https://www.npmjs.com/>